

Procurement Procedure EMSA/OP/08/2014

Questions/ Answers

Question 01 (dated 18/07/2014, 12:15):

Considering the geographical constraint of personally reviewing the document, I request you to provide us the following details before we buy the document:

- 1) List of Items, Schedule of Requirements, Scope of Work, Terms of Reference, Bill of Materials required.
- 2) Soft Copy of the Tender Document through email.
- 3) Names of countries that will be eligible to participate in this tender.
- 4) Information about the Tendering Procedure and Guidelines
- 5) Estimated Budget for this Purchase
- 6) Any Extension of Bidding Deadline?
- 7) Any Addendum or Pre Bid meeting Minutes?

Answer to question 01:

All tender documentation can be found on the Internet at <http://emsa.europa.eu/work/procurement/calls.html> in the procurement section related to the tender EMSA/OP/08/2014 and can be downloaded free of charge.

Question 2 (dated 07/08/2014 20:24):

1. Can you please confirm that the duration of the contract is 4 years?
2. Can you provide an historical log of the maintenance requests?
3. In the table of point 12 of the Tender Specification, the profile "Application Developer" is duplicated. Is it correct our understanding?
4. The items referred in the Tenderer's Checklist as "For the proposed methods and approach to implement the contract requirements to perform the tasks and supporting tools" should be described in Part C ou Part D of the proposal?

Answers to question 02:

1. This is the maximum duration of the contract; the allocated budget needs to be spent within 4 years from the contract signature. The budget could be spent in a lesser amount of time in which case a new procurement procedure might be launched.

2. The maintenance for the current SharePoint at EMSA started only in January 2014. Since then this is the number of hours booked (until and including July 2014):
Senior product specialist: 12 hours
Developer: 160 hours
3. This is a clerical error; the rate for the 'Application Developer' profile needs to be provided only once.
4. The items referred to in the Tender's Checklist as "For the proposed methods and approach to implement the contract requirements to perform the tasks and supporting tools" should be described in Part C of the proposal.

Question O3 (dated 14/08/2014 00:39):

Following our previous request clarification, we would like to rephrase our question:

In order to achieve a very strong proposal and a high excellence team with a very broad set of skills, we're evaluating a joint offer /subcontracting with other entity. Our question is, considering that we want EMSA to evaluate our proposal, regarding technical selection criteria (level of partnerships, references, human resources, etc.), with assets from both companies, if we can subcontract that company (and the sum of the competencies of both companies are evaluated for selection criteria) or if we must present a joint offer.

Still regarding the Tender EMSA/OP/09/2014, if the tenderer relies on its subcontractor's capacities for technical selection criteria, the tenderer has to submit, for each subcontractor, any of the following documents?

- Signed Cover letter of each subcontractor
- LEF of each subcontractor
- Financial Form of each subcontractor
- Declaration on honour of each subcontractor
- Financial Statements of each subcontractor
- Statement of overall turnover of each subcontractor

Answers to question O3:

Tenderers may submit offers individually or with subcontractor(s) or as grouping presenting one joint offer.

If the tenderer intends to either sub contract part of the work or develop the work in co-operation with other partners (Joint Offers), he shall indicate this in his offer by completion of the form – "Information regarding joint offers and subcontracting".

In case of subcontracting and in case of joint offers the exclusion criteria and the selection criteria for "economic and financial capacity" will be assessed in

relation to each economic operator individually. Concerning the selection criteria for “technical and professional capacity”, the evidence provided will be checked to ensure that the tenderer and its subcontractors as a whole fulfil the criteria.

Regarding the second part of your question, on documents needed when tenderer provides bid with subcontractors, please note that the tenderer needs to submit:

- “Information regarding joint offers and subcontracting form”
- Declaration on honor of each subcontractor
- Financial Statements of each subcontractor
- Statement of overall turnover of each subcontractor
- Any other documents necessary to evaluate technical and professional capacity.

Question 04 (dated 13/08/2014 17:13):

“I have consulted the last version of the question and answers document, from the tender specified in the subject of this e-mail, and there is an answer saying that:

“The items referred to in the Tender’s Checklist as “For the proposed methods and approach to implement the contract requirements to perform the tasks and supporting tools” should be described in Part C of the proposal.”

I’m confused about this answer because in point 11 (from the Tender Specification document) it is said that Part D shall have all information request in point 15.1 and it’s that point, in my understanding, that requests the following documents:

- Project approach overview
- Project management methodology
- Plan to acquire knowledge about the system
- Design methodologies and tools
- Software development methodology and supporting tools
- Testing methodology
- Load and stress tests approach and supporting tools
- Response times to EMSA requests, to be provided in Appendix 4 – Response times

Henceforth, if my understanding is correct, then all this information shall be specified in Part D (instead of Part C) of our proposal.”

Answers to question 04:

EMSA will consider the tenderer’s response to this question regardless if it is in Part C or in Part D, on the condition that the section is clearly labelled as such. We recommend tenderers to indicate for each item on the checklist in what section of their proposal it is covered.

Question 05 (dated 20/08/2014 15:00):

"Dear Sir or Madam, please find enclosed a few questions on the above mentioned tender procedure.

Question on Appendix 04 (incident resolution time):

In the area of software development and maintenance it is not possible to guarantee an incident resolution time as requested in Appendix 04. Especially incidents caused by a cumulative system update by the manufacturer itself can vary heavily in time until they are solved. This is outside the sphere of influence of the bidder.

The bidders can only guarantee to come back to EMSA with a qualified answer within a certain time. We kindly ask you to change Appendix 04. It might be phrased "Time in which EMSA can expect a qualified answer for the reported problem (in hours)" instead of "Time in which EMSA can expect a solution for the reported problem (in hours)

Question on Appendix 05 – CV template Application Tester:

Experience in working with NUnit, Selenium is not a common requirement for an Application Tester in a SharePoint environment. As we would like to offer professional SharePoint Application Tester, we kindly ask you to change this requirement from "mandatory" to "desirable". If not please describe in detail for which purposes you need this skill.

Question on tendering specifications:

Could you please explain in more detail what is meant by "Design methodologies and tools"? Is it pointing towards software design or to design regarding wireframe, user experience and user acceptance?"

Answers to question 05:

Question on Appendix 04 (incident resolution time):

Please note that EMSA is not in a position to change the content of the Appendix 04 above at this advanced stage of an Open call for tender. Furthermore please note that EMSA provides development and quality environments for the specific purpose of verifying that any patches, whether they are created by the contractor or the manufacturer, will not have any negative impact on the production environment. On top of that the contractor should prepare a rollback plan for each software deployment and/or configuration change. Also consider that the tenderer is free to specify the resolution times he is comfortable with. We therefore feel that receiving a 'qualified answer' to an incident report leaves EMSA potentially exposed to unacceptably long incident resolution times whilst the contractor has sufficient means to ensure he complies with the quoted incident resolution times.

Question on Appendix 05 – CV template Application Tester:

Please note that EMSA is not in a position to change the content at this advanced stage of an Open call for tender. Experience shows that relying on the manual execution of test scripts is neither cost efficient nor time efficient. As much of the contract deals with custom SharePoint development we consider the requirement for the testers to have experience with automated testing tools is a reasonable requirement. We are willing to consider experience with other, similar tools on the condition that these tools are available to EMSA at no extra cost, during and after

the execution of the framework contract. In this case we require that automated testing for both the User Interface and the code developed for EMSA can be covered. The automated code testing tool shall also be able to integrate with EMSA's continuous integration environment based on Hudson.

Question on tendering specifications:

Software design, design regarding wireframe, user experience and user acceptance all need to be addressed in the reply to the tender.

Question 06 (dated 22/08/2014 09:18):

"Dear Sir/Madam,

as we're preparing our tender for your SharePoint Collaboration Platform we've found one problem in Annex 2: could you please explain us your understanding of the term "Agile Methodology"? Because our understanding is little bit different, probably...."

Answers to question 06:

What EMSA project Team understands by the term 'Agile Methodology' is based on the principles posed in Agile Manifesto (<http://agilemanifesto.org/>):

"We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:

Individuals and interactions over processes and tools; Working software over comprehensive documentation; Customer collaboration over contract negotiation; Responding to change over following a plan.

That is, while there is value in the items on the right, we value the items on the left more."

Futhermore Agile software development is a group of software development methods in which requirements and solutions evolve through collaboration between self-organizing, cross-functional teams. It promotes adaptive planning, evolutionary development, early delivery, continuous improvement and encourages rapid and flexible response to change. It is a conceptual framework that focuses on delivering working software with the minimum amount of work.

This should not be confused by an anti-methodology, anti-process approach, as already noted by Jim Highsmith in the introduction of the Agile Manifesto: "The Agile movement is not anti-methodology, in fact, many of us want to restore credibility to the word methodology. We want to restore a balance. We embrace modeling, but not in order to file some diagram in a dusty corporate repository. We embrace documentation, but not hundreds of pages of never-maintained and rarely-used tomes. We plan, but recognize the limits of planning in a turbulent environment. Those who would brand proponents of XP or SCRUM or any of the other Agile Methodologies as "hackers" are ignorant of both the methodologies and the original definition of the term hacker."

The Agile Manifesto is based on twelve principles:

1. Customer satisfaction by rapid delivery of useful software
2. Welcome changing requirements, even late in development
3. Working software is delivered frequently (weeks rather than months)
4. Close, daily cooperation between business people and developers
5. Projects are built around motivated individuals, who should be trusted
6. Face-to-face conversation is the best form of communication (co-location)
7. Working software is the principal measure of progress
8. Sustainable development, able to maintain a constant pace
9. Continuous attention to technical excellence and good design
10. Simplicity—the art of maximizing the amount of work not done—is essential
11. Self-organizing teams
12. Regular adaptation to changing circumstances

There are many specific agile development methods. Most promote development, teamwork, collaboration, and process adaptability throughout the life-cycle of the project.

Iterative, incremental and evolutionary

Most agile methods break tasks into small increments with minimal planning and do not directly involve long-term planning. Iterations are short time frames (timeboxes) that typically last from one to four weeks. Each iteration involves a cross-functional team working in all functions: planning, requirements analysis, design, coding, unit testing, and acceptance testing. At the end of the iteration a working product is demonstrated to stakeholders. This minimizes overall risk and allows the project to adapt to changes quickly. An iteration might not add enough functionality to warrant a market release, but the goal is to have an available release (with minimal bugs) at the end of each iteration. Multiple iterations might be required to release a product or new features.

Efficient and face-to-face communication

No matter what development disciplines are required, each agile team will contain a customer representative, e.g. Product Owner in Scrum. This person is appointed by stakeholders to act on their behalf and makes a personal commitment to being available for developers to answer mid-iteration questions. At the end of each iteration, stakeholders and the customer representative review progress and re-evaluate priorities with a view to optimizing the return on investment (ROI) and ensuring alignment with customer needs and company goals.

In agile software development, an information radiator is a (normally large) physical display located prominently in an office, where passers-by can see it. It presents an up-to-date summary of the status of a software project or other product. The name was coined by Alistair Cockburn, and described in his 2002 book *Agile Software Development*. A build light indicator may be used to inform a team about the current status of their project.

Very short feedback loop and adaptation cycle

A common characteristic of agile development are daily status meetings or "stand-ups", e.g. Daily Scrum (Meeting). In a brief session, team members report to each other what they did the previous day, what they intend to do today, and what their roadblocks are.

Quality focus

Specific tools and techniques, such as continuous integration, automated unit testing, pair programming, test-driven development, design patterns, domain-driven design, code refactoring and other techniques are often used to improve quality and enhance project agility.

Compared to traditional software engineering, agile development is mainly targeted at complex systems and projects with dynamic, undeterministic and non-linear characteristics, where accurate estimates, stable plans and predictions are often hard to get in early stages, and big up-front designs and arrangements will probably cause a lot of waste, i.e. are not economically sound. These basic arguments and precious industry experiences learned from years of successes and failures have helped shape Agile's favor of adaptive, iterative and evolutionary development.

Philosophy

Compared to traditional software engineering, agile development is mainly targeted at complex systems and projects with dynamic, undeterministic and non-linear characteristics, where accurate estimates, stable plans and predictions are often hard to get in early stages, and big up-front designs and arrangements will probably cause a lot of waste, i.e. are not economically sound. These basic arguments and precious industry experiences learned from years of successes and failures have helped shape Agile's favor of adaptive, iterative and evolutionary development.

Adaptive vs. Predictive

Development methods exist on a continuum from adaptive to predictive. Agile methods lie on the adaptive side of this continuum. One key of adaptive development methods is a "Rolling Wave" approach to schedule planning, which identifies milestones but leaves flexibility in the path to reach them, and also allows for the milestones themselves to change. Adaptive methods focus on adapting quickly to changing realities. When the needs of a project change, an adaptive team changes as well. An adaptive team will have difficulty describing exactly what will happen in the future. The further away a date is, the more vague an adaptive method will be about what will happen on that date. An adaptive team cannot report exactly what tasks they will do next week, but only which features they plan for next month. When asked about a release six months from now, an adaptive team might be able to report only the mission statement for the release, or a statement of expected value vs. cost.

Predictive methods, in contrast, focus on analysing and planning the future in detail and cater for known risks. In the extremes, a predictive team can report exactly what features and tasks are planned for the entire length of the development process. Predictive methods rely on effective early phase analysis and if this goes very wrong, the project may have difficulty changing direction. Predictive teams will often institute a Change Control Board to ensure that only the most valuable changes are considered.

Iterative vs. Waterfall

One of the differences between agile and waterfall is that testing of the software is conducted at different stages during the software development lifecycle. In the Waterfall model, there is always a separate testing phase near the completion of an implementation phase. However, in Agile and especially Extreme programming, testing is usually done concurrently with coding, or at least, testing jobs start in early iterations.

Code vs. Documentation

The agile principle of "Working software over comprehensive documentation" should not be misunderstood as "We want to spend all our time coding". It does however recognise that huge amounts of static documentation are often out-of-date by the time they have been created. For more documentation on our proposed approach to documentation for code documentation please read 'Clean Code: A Handbook of Agile Software Craftsmanship' by Robert C. Martin.

Further information can be found here:

http://en.wikipedia.org/wiki/Agile_software_development

Question 07 (dated 25/08/2014 11:20):

"Dear Sir or Madam, please find enclosed a question on the above mentioned tender procedure.

Question on Appendix 04:

Are we right in assuming that you only refer to „all services“ which were developed by the contract holder himself and not by any other third party?

Answers to question 07:

This assumption is INCORRECT:

Your assumption is INCORRECT:

- this tender is for the further development and support of an existing Sharepoint-based application
- only one contractor will be involved in that support
- therefore, if all services are unavailable, the contractor will be expected to find a solution
 - even if the incident involves services, which they did not develop
- the same logic also applies for 'critical' and 'standard' maintenance types as well as 'urgent':
 - all services must be supported

Published on 26/08/2014

Requests for additional information regarding this tender should be sent by e-mail to the following address OPEN082014@emsa.europa.eu. Requests for additional information received less than five working days before the closing date for submission of tenders will not be processed.

The deadline for submission of the bids of this tender is 01/09/2014.

Responsibility for monitoring the Agency's website for replies to queries and/or further information remains with potential applicants.